arXiv:2303.02626v1 [stat.ME] 5 Mar 2023

贝叶斯广义加性模型

Antti Solonen **Stratos Staboulis**

2023年3月7日

摘要

广义加性模型（广义加性模型）提供了一种将参数化技术与非参数（函数逼近）技术相结合的方法，因此成为适用于众多建模问题的灵活工具。例如，广义加性模型可用于为标准线性回归模型引入灵活性，以表达某种现象的“近似线性”行为。在物理模型中也常常需要使用广义加性模型：理论给出的模型只是对现实的近似，而人们希望将系数表示为函数，而不是常数。本文从贝叶斯视角讨论广义加性模型，重点关注线性加性模型，其中最终模型可以表述为线性高斯系统。我们讨论使用高斯过程和局部基函数方法描述广义加性模型中的未知函数，以及为这些函数指定先验分布的技术，其中包括空间变化的平滑度。本文还讨论同时包含单变量函数和多变量函数的广义加性模型。为缓解与广义加性模型相关的调参问题，本文介绍了超参数估计技术。文中讨论的所有示例均提供了实现代码。[1]

1 引言与符号说明

让我们考虑形式为以下形式的广义加性模型。

$$y_i = f_1(\mathbf{x}_{i,1}, \boldsymbol{\theta}_1) + f_2(\mathbf{x}_{i,2}, \boldsymbol{\theta}_2) + \cdots + f_M(\mathbf{x}_{i,M}, \boldsymbol{\theta}_M) + \varepsilon_i, \tag{1}$$

其中 f_i 为函数， y_i 为观测值，$\mathbf{x}_{i,j}$ 为函数 j所需的输入观测值， $\boldsymbol{\theta}_i$ 为函数 f_i 的一些未知参数， ε_i 为测量误差。

此类模型的一个动机，是能够将非参数效应纳入标准线性回归模型；上面的某些项可能只是 $f_j(\mathbf{x}_{i,j}, \boldsymbol{\theta}_j)$ $_i = x_{i,j}\theta_j$，而另一些项则可能被视为未知函数，其中参数例如可以是该函数在一组指定输入点处的未知取值。另一个动机是“变系数模型”，在这类模型中，线性回归模型的系数被建模为未知函数：

$$y_i = f_1(\mathbf{z}_{i,1}, \boldsymbol{\theta}_1)x_{i,1} + f_2(\mathbf{z}_{i,2}, \boldsymbol{\theta}_2)x_{i,2} + \cdots + f_M(\mathbf{z}_{i,M}, \boldsymbol{\theta}_M)x_{i,1} + \varepsilon_i, \tag{2}$$

其中，$\mathbf{z}_{i,j}$ 是函数 f_j 对观测值 i进行建模所需的输入。使用上述类型的模型，可以将响应变量与输入变量之间的关系表示为“近似线性”，或者将回归系数表示为“近似恒定”，因此，该模型为标准线性回归模型提供了一个灵活的框架。

请注意，上述模型与典型的 GAM 表述略有不同；在典型表述中，观测值附带一个非线性链接函数 $g(y_i)$。在此处，我们忽略这种非线性（将情况限制为 $g(y_i) = y_i$），仅关注线性加性模型，不过论文中讨论的许多思想同样可以推广到拟合此类广义加性模型。

1.1 一般线性-高斯模型

在本文中，我们考虑 模型，其中函数 f_i 使得关于未知 n
参数 $\boldsymbol{\theta} = [\boldsymbol{\theta}_i, \boldsymbol{\theta}_2 \quad \cdots, \boldsymbol{\theta}_M]^T$是线性的（但相对于模型输入可能是非线性的）。此外，w e

[1] https://github.com/solbes/gam_paper_examp les

希望通过高斯先验约束未知参数值。也就是说，用于估计 $\boldsymbol{\theta}$ 的整个模型始终可以写成以下形式

$$\mathbf{y} = \mathbf{A}\boldsymbol{\theta} + \boldsymbol{\varepsilon} \tag{3}$$

$$\boldsymbol{B\theta} \sim \mathrm{N}\left(\boldsymbol{\mu}_{\mathrm{pr}}, \boldsymbol{\Gamma}_{\mathrm{pr}}\right), \tag{4}$$

其中$\mathbf{y} = [y_1, \cdots, y_N]^T$是所有观测值组成的向量。此外，假设存在高斯测量误差，$\boldsymbol{\varepsilon} \sim \mathrm{N}\left(000, \boldsymbol{\Gamma}_{\mathrm{obs}}\right)$，我们得到一个线性高斯系统，其后验分布为 $\boldsymbol{\theta}|\mathbf{y} \sim \mathrm{N}\left(\boldsymbol{\mu}_{\mathrm{pos}}, \boldsymbol{\Gamma}_{\mathrm{pos}}\right)$，其中

$$\boldsymbol{\Gamma}_{\mathrm{pos}}^{-1} = \mathbf{A}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{obs}}^{-1}\mathbf{A} + \mathbf{B}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{pr}}^{-1}\mathbf{B} \tag{5}$$

$$\boldsymbol{\mu}_{\mathrm{pos}} = \boldsymbol{\Gamma}_{\mathrm{pos}}\left(\mathbf{A}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{obs}}^{-1}\mathbf{y} + \mathbf{B}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{pr}}^{-1}\boldsymbol{\mu}_{\mathrm{pr}}\right). \tag{6}$$

请注意，$\mathbf{A}$ 和$\mathbf{B}$ 通常是稀疏矩阵，噪声协方差通常取对角形式，因此即使对于高维数据和参数，也可以使用稀疏线性代数方法高效地求解上述系统。此外，后验精度矩阵 $\boldsymbol{\Gamma}_{\mathrm{pos}}^{-1}$ 通常也具有稀疏性，可用于高效地从高斯后验分布中进行抽样。

本文后续将进一步讨论从 f_i 的各种表述到上述系统的映射。例如，将 f_j设置为$(\mathbf{x}_{i,j}, \boldsymbol{\theta}_j)_i = x_{i,j}\theta_j$即可得到经典线性回归，此时我们有

$$\boldsymbol{A} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_M \end{bmatrix}, \tag{7}$$

其中 $\mathbf{x}_j = [x_{1,i},\ x_{2,j},\ \cdots,\ x_{N,j}]^T$ 是线性回归中典型的设计矩阵 .

先验系统中的线性变换 $\boldsymbol{B}$ 适用于这样的情况：我们不想直接对未知量施加惩罚，而是例如对空间网格中连续函数值之间的差异施加惩罚。请注意，$\boldsymbol{B}$ 不必是满秩的；只要上述后验分布是适定的，先验就可以是不适定的。例如，如果我们只对连续函数值之间的差异设置先验，而不直接对函数值施加惩罚，就属于这种情况。

1.2 基函数表示

在广义加性模型中，我们通常希望使用非参数技术来估计函数 f_i。为此，在大多数情况下，我们可以用选定基函数的线性组合来表示这些函数:

$$(f_i)_j = \sum_{k=1}^{K_i} \alpha_{i,k} p_{i,k}(\mathbf{x}_{i,j}), \tag{8}$$

其中，与该函数相关的未知参数是权重 $\boldsymbol{\theta}_i = [\alpha_1, ..., \alpha_K]$。例如，基函数可以是一组在选定节点处定义的样条函数。

借助上述未知函数的基函数表示，我们得到一个“堆叠形式”的线性系统，其中每个 f_i 会向系统矩阵$\mathbf{A}$引入 K_i 列:

$$\mathbf{A} = \begin{bmatrix} \overbrace{p_{1,1}(\mathbf{x}_{1,1}) \ \cdots \ p_{1,K_1}(\mathbf{x}_{1,1})}^{f_1} & \overbrace{p_{2,1}(\mathbf{x}_{2,1}) \ \cdots \ p_{2,K_2}(\mathbf{x}_{2,1})}^{f_2} & \cdots & \overbrace{p_{M,1}(\mathbf{x}_{M,1}) \ \cdots \ p_{M,K_M}(\mathbf{x}_{M,1})}^{f_M} \\ p_{1,1}(\mathbf{x}_{1,2}) \ \cdots \ p_{1,K_1}(\mathbf{x}_{1,2}) & p_{2,1}(\mathbf{x}_{2,2}) \ \cdots \ p_{2,K_2}(\mathbf{x}_{2,2}) & \cdots & p_{M,1}(\mathbf{x}_{M,2}) \ \cdots \ p_{M,K_M}(\mathbf{x}_{M,2}) \\ \vdots & \vdots & \vdots & \vdots \\ p_{1,1}(\mathbf{x}_{1,N}) \ \cdots \ p_{1,K_1}(\mathbf{x}_{1,N}) & p_{2,1}(\mathbf{x}_{2,N}) \ \cdots \ p_{2,K_2}(\mathbf{x}_{2,N}) & \cdots & p_{M,1}(\mathbf{x}_{M,N}) \ \cdots \ p_{M,K_M}(\mathbf{x}_{M,N}) \end{bmatrix} \tag{9}$$

有时，将 平移添加到基函数表示中是有意义的，在这种情况下，我们将 e

$$(f_i)_j = m_i(\mathbf{x}_{i,j}) + \sum_{k=1}^{K_i} \alpha_{i,k} p_{i,k}(\mathbf{x}_{i,j}), \tag{10}$$

其中，m_i 是函数 i的平移；每个 m_i 都是输入数据的已知函数。在这种情况下，$\mathbf{A}$ 矩阵与上述未平移版本中的保持不变，但在模型拟合之前，需要从观测数据中减去这些平移量，以得到式（3）—（4）中给出的系统的标准形式。

另请注意，上述内容可以直接扩展到方程（2）中的“变系数模型”：只需将基向量与相应的输入数据相乘即可。

2 文献综述

书籍 [1] 或许是关于这一主题最为全面的著作，此外，近期论文 [2] 还讨论了广义加性模型的一些最新发展。它们涵盖了广义加性模型的各个方面，并介绍了用于计算的 R 代码。该书主要从频率学派视角撰写，但也简要讨论了贝叶斯视角，例如阐述了惩罚平滑样条与高斯平滑先验之间的联系。

文献 [1, 2] 中的广义加性模型侧重于通过平滑样条描述的局部基函数。本文讨论局部和全局基函数，并介绍一些新颖特性，例如空间变化的平滑度、降维，以及平滑先验的任意阶数（在以往的工作中，阶数始终为二）。此外，我们还介绍用于估计广义加性模型超参数的贝叶斯方法。

皇家统计学会的论文 [3] 讨论了使用 INLA（积分嵌套拉普拉斯近似）拟合 GAM 模型。该论文探讨了超参数估计、各种非高斯似然，以及如何高效地近似后验分布。高斯马尔可夫随机场（GRMF）被选作先验。所引论文主要关注非高斯问题中的推断方法，而本文则将范围限定在线性高斯情形，并着重讨论如何通过各种局部和全局基函数构造来表示广义加性模型。

多元自适应回归样条（MARS），参见 [4], , 是构建广义加性模型的另一种方法。在 MARS 中，基函数要么是常数，要么是“铰链函数”；该方法还可通过铰链函数的乘积扩展到多个维度。此方法首先以贪心方式添加项，然后通过交叉验证移除项来剪枝模型。与本文不同的是，这种方法并非采用贝叶斯视角。此外，我们认为，在 MARS 方法中，要控制模型的光滑性及其他属性（例如周期性）相当困难，而这正是本文的主要议题之一。

3 基于高斯过程的全局基函数

高斯过程是一种常用的非参数函数逼近技术。高斯过程（GP）是一个无限维（函数空间）概念，粗略地说，可以将其定义为这样一种随机过程：任意有限个实现对应的函数值集合，其分布均为多元正态分布。这里我们在广义加性模型的语境下讨论高斯过程；有关高斯过程的一般性全面介绍，请参见例如 [5]。

GP 由其均值函数 $m(x)$ 和协方差函数（或核函数） $k(x,\ x')$ 表征。为便于实际计算，可以将 GP 离散化到所选的输入值网格上，此时 GP 就转化为一个多元正态分布。

这里，我们将 GAM 模型中的函数表示为高斯过程：

$$f_i(\mathbf{x}, \boldsymbol{\theta}) \sim \mathcal{GP}\left(m_i(\mathbf{x}), k_i(\mathbf{x}, \mathbf{x}')\right). \tag{11}$$

模型参数 $\boldsymbol{\theta}$ 的定义将在下文进一步说明。实际应用中，我们将函数离散化到固定的输入点网格上，因此 $\mathbf{f}_i \sim N(\boldsymbol{\mu}_{f_i},\ \boldsymbol{\Sigma}_{f_i})$，其中均值向量和协方差矩阵由所选的均值函数和协方差函数计算得出。

为了在此处将基函数表示用于广义加性模型，我们对协方差矩阵进行特征值分解：

$$\boldsymbol{\Sigma}_f = \sum_{i=1}^{n} \lambda_i \mathbf{q}_i \mathbf{q}_i^{\mathrm{T}} = \mathbf{P}\mathbf{P}^{\mathrm{T}}, \tag{12}$$

矩阵 $\mathbf{P}$ 的第 i 列是第 i 个特征向量乘以对应的特征值：$\mathbf{p}_i = \sqrt{\lambda_i}\mathbf{q}_i$。容易验证，$\mathbf{f}_i = \boldsymbol{\mu}_{f_i} + \mathbf{P}\overline{\boldsymbol{\theta}}$，其中 $\boldsymbol{\theta} \sim N(\mathbf{0},\ \mathbf{I})$。此时，基函数是 $\mathbf{P}$ 的各列，模型参数 $\boldsymbol{\theta}$ 是基向量的权重（采用方便的独立同分布高斯先验），平移由 $\boldsymbol{\mu}_{f_i}$给出，随后我们便可使用第 1.2 节讨论的基向量表示进行计算。不过请注意，此处的函数定义在输入点网格上，而实际应用中需要在观测位置处计算函数值。只需对这些位置进行插值即可。

从无限维的角度来说，我们的目标是将特征函数用作此处广义加性模型的基向量。上文所述的离散化-插值技巧，是在实践中近似特征函数的一种方法。不过，还有其他方法，例如 Nyström 方法，参见 [6, 7]。此外，对于某些核函数，我们或许可以直接得到其解析形式的特征函数 [5]。

采用这种方法，得到的基向量数量（以及新增未知量的数量）等于为高斯过程表示所选取的网格点数量。不过，通过剔除对协方差贡献不大的基向量，可以直接降低估计问题的维度。我们可以选择

仅保留前 $k < n$ 个特征值，并近似为 $\boldsymbol{\Sigma}_f \approx \mathbf{P}_k \mathbf{P}_k^{\mathrm{T}}$，其中$\mathbf{P}_k$包含与 k 个最大特征值对应的基向量，从而使截断协方差包含原始协方差的例如 99.99%[2]。现在，我们可以相应地重新参数化：$\mathbf{f} = \boldsymbol{\mu}_f + \mathbf{P}_k \boldsymbol{\theta}$，并且只估计 k 个权重，而不是原来的 n 个。这是一种众所周知的降维方法（"*truncated SVD*"方法），常用于反问题，而反问题通常涉及高维估计问题；例如，当未知量是空间函数时，参见 [6]。降维技术的成功取决于所选先验高斯过程的平滑程度；假设未知函数越平滑，估计所需的维度就越低。当（相互独立的）输入变量数量大于 1 时，如下一节所述，这种方法尤其有用。

3.1 多维高斯过程基

由于我们在选定的网格上表示高斯过程，未知量的总数会随着输入空间维度的增加而迅速增长。例如，若在三元高斯过程的每个方向上使用50个网格点，则需要 $50^3 = 125000$个点。甚至连存储如此高维的协方差矩阵都不可能。此外，利用稀疏矩阵代数通常也很困难，因为通常使用的大多数协方差核都会产生稠密的协方差矩阵。

将范围限制为可分离协方差函数，并结合上一节所述的降维方法，可以提供一种解决途径。如果一个协方差函数可以表示为若干低维协方差函数的乘积，则称其为可分离的。例如，二维可分离协方差函数可写为$k(\mathbf{x},\mathbf{x}') = k(x_1, x_1')k(x_2, x_2')$。现在，我们将两个变量离散化，使得变量 x_1 有 n 个网格点，变量 x_2有 m 个网格点，并计算一维协方差矩阵$\mathbf{K}_1 \in \mathbb{R}^{n\times n}$ 和$\mathbf{K}_2 \in \mathbb{R}^{m\times m}$。现在，联合空间中的完整协方差矩阵可通过各个协方差矩阵的克罗内克积计算得到（假设变量采用适当的排序）：

$$\mathbf{K} = \mathbf{K}_1 \otimes \mathbf{K}_2 \in \mathbb{R}^{mn\times mn}. \tag{13}$$

联合空间中的基向量又如何呢？我们可以利用克罗内克积的性质（例如，参见 [8], ），通过低维矩阵的分解来计算整个矩阵的特征分解，而无需显式构造$\mathbf{K}$。在二维示例中，若一维协方差的分解分别为$\mathbf{K}_1 = \mathbf{Q}_1\boldsymbol{\Lambda}_1\mathbf{Q}^{\mathrm{T}}$ 和$\mathbf{K}_2 = \mathbf{Q}_2\boldsymbol{\Lambda}_2\mathbf{Q}_2^{\mathrm{T}}$，则二维协方差的分解为

$$\mathbf{K} = \mathbf{K}_1 \otimes \mathbf{K}_2 = (\mathbf{Q}_1 \otimes \mathbf{Q}_2)(\boldsymbol{\Lambda}_1 \otimes \boldsymbol{\Lambda}_2)(\mathbf{Q}_1 \otimes \mathbf{Q}_2)^{\mathrm{T}}. \tag{14}$$

也就是说，特征向量是各个特征向量的克罗内克积，特征值则是各个特征值的乘积（对向量和值进行适当排序后）。对于高于二维的情形，分解过程也类似。

这样，我们就可以针对每个输入变量应用降维技术，只保留对先验协方差有显著贡献的特征向量，同时舍弃贡献不显著的特征向量。此过程可以大幅降低估计未知量联合空间的整体维度。

3.2 一些有用的核函数及示例

在广义加性模型语境中，或许最有用的核函数是平方指数核

$$k(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp\left(-\frac{d(\mathbf{x}, \mathbf{x}')^2}{2L^2}\right), \tag{15}$$

其中 $d(\mathbf{x},\mathbf{x}')$ 是欧几里得距离， σ^2 是方差， L 是相关长度。方差大致描述函数相对于均值可以波动的幅度，而相关长度决定函数的光滑性，换句话说，决定相距一定距离的两个函数值有多相似。该核函数产生的实现非常光滑，其特征值衰减很快，因此适用于上文讨论的降维方法。

有时我们需要描述周期函数。例如，处理角度或时间周期的函数时。平方指数核的周期版本可以写作

$$k_{\mathrm{periodic}}(\mathbf{x}, \mathbf{x}') = \sigma^2 \exp\left(\frac{-2\sin^2(\pi d(\mathbf{x}, \mathbf{x}')/\phi)}{L^2}\right), \tag{16}$$

其中 ϕ 是周期。此外，我们通常可以假设底层函数具有对称性； $f(x) = f(-x)$。这类对称核函数可以通过已有核函数 $k(x, x')$ 构造：

$$k_{\mathrm{sym}}(x, x') = k(x, x') + k(-x, x'). \tag{17}$$

[2]例如，可以选取满足 $(\sum_{i=1}^{k}\lambda_i)/(\sum_{i=1}^{n}\lambda_i) < 0.9999$ 条件的特征向量 k 。

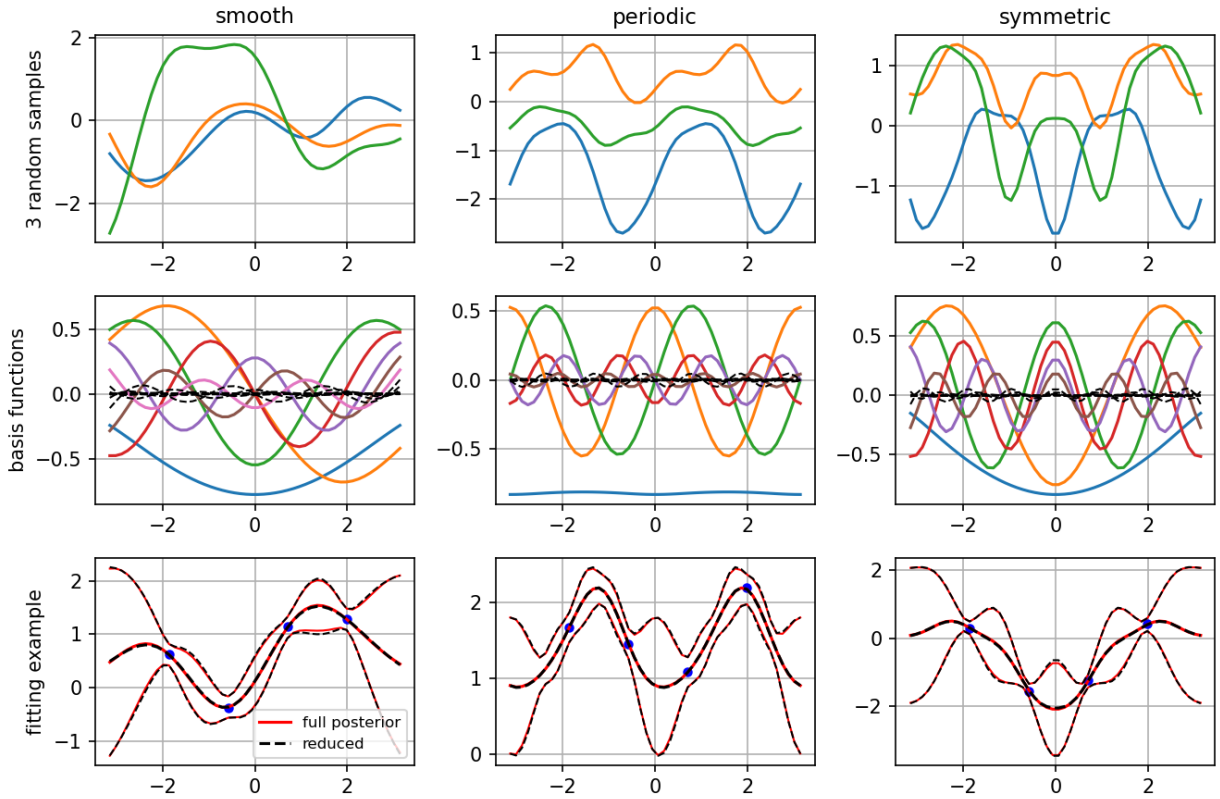

图1：平方指数核（第一列）及其周期版本（中间列）和对称版本（最后一列）的随机样本（第一行）、基向量（第二行）和简单拟合示例（第三行）。第二行图中的黑色虚线展示了估计中被剔除的基函数。最后一行比较了使用完整基和缩减基计算得到的均值及 $\pm$两个标准差。

图 1 给出了所讨论的核函数及其在降秩基函数中的应用示例。可以看到，采用具有强平滑作用的核函数进行降维，只需少量基向量即可对函数进行建模。

4 局部基函数

迄今为止，最常见的广义加性模型表述方式是使用样条以非参数方式表示未知函数；例如，参见 [1]，其中对广义加性模型中的样条进行了全面论述。基于样条的广义加性模型中，未知量是样条基函数的系数。

函数的光滑性可以通过两种机制实现：1）选择的节点位置数量（节点数量越少，函数越光滑）；2）显式惩罚函数的光滑性，例如为二阶导数添加惩罚项：$\lambda \int f''(x)dx$，其中 λ 是一个调节参数（超参数）。样条的导数可以通过解析方式计算，由此得到针对未知样条系数的二次惩罚。用贝叶斯术语来说，这等价于对样条系数设置高斯先验。

基于样条的广义加性模型所产生的估计问题可能具有高维性。例如，通常选择一个“稍微过于精细”的节点位置网格是合理的，然后利用光滑性惩罚为函数选择合适的光滑程度。不过，由于样条基函数仅在相应节点位置附近非零，在其他位置均为零，因此可以在计算中使用稀疏线性代数，从而缓解这一难题。

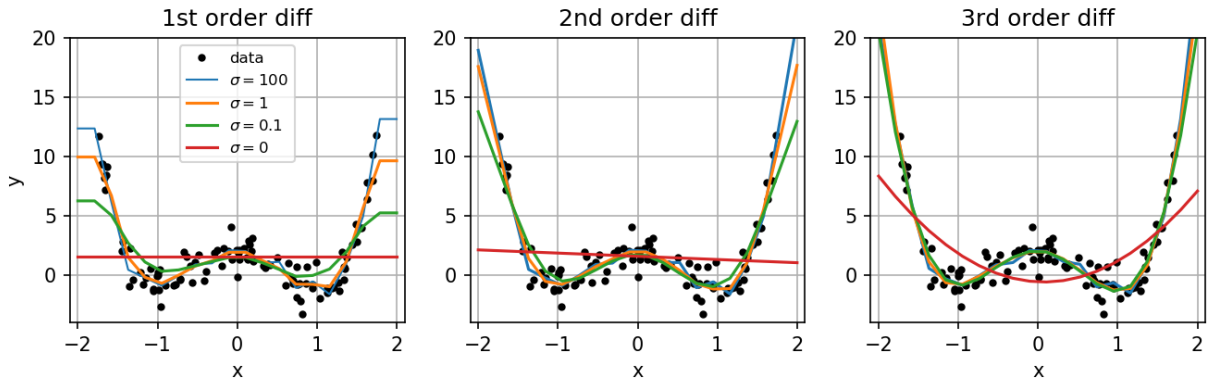

图 2：围绕真实函数 $f(x) = 3x^4 - 6x^2 + 2$ 生成的数据拟合结果，使用了三种不同阶数的差分先验和四种不同的先验方差。

4.1 一维中的一阶样条与差分先验

这里我们来讨论作为 GAM 构建模块的一阶 B样条。这等同于直接将函数值作为未知量，并在选定输入点之间的区域进行线性插值。在我们的估计框架中，这意味着矩阵$\mathbf{A}$ 是一个稀疏线性插值矩阵，它将选定输入网格上的给定函数值映射到观测位置。下面给出了$\mathbf{A}$ 的一个示例：

$$\mathbf{A} = \begin{bmatrix} 0 & \cdots & w_1 & 1-w_1 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & w_2 & 1-w_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}, \tag{18}$$

其中， w_i 是输入观测值与其超过的第一个输入网格值之间的距离。因此，使用输入网格点 $\mathbf{x}_g = [x_{g,1}, x_{g,2}, ..., x_{g,N}]$ 时，矩阵元素可表示为

$$A_{ij} = \begin{cases} x_i - x_{g,j} & \text{if } x_i \in [x_{g,j}, x_{g,j+1}] \\ 1 - (x_i - x_{g,j}) & \text{if } x_i \in [x_{g,j-1}, x_{g,j}] \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

这里不妨拓宽视角，而不只关注带有单个调节参数的二阶光滑性惩罚。当然，我们也可以直接对函数值施加高斯先验。但很多时候，对函数值的变化（导数）进行惩罚更有用。在选定的网格上近似导数，可得到稀疏差分矩阵。差分矩阵的阶数不一定非得是二阶，尽管基于样条的广义加性模型通常采用二阶差分。下面给出长度为 5 的一维输入网格对应的一阶、二阶和三阶差分矩阵示例：

$$\mathbf{D}_1 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \quad \mathbf{D}_2 = \begin{bmatrix} -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \end{bmatrix} \quad \mathbf{D}_3 = \begin{bmatrix} -1 & 3 & -3 & 1 & 0 \\ 0 & -1 & 3 & -3 & 1 \end{bmatrix}. \tag{20}$$

请注意，这里我们忽略差分算子的“边界条件”，只对差分进行惩罚，而不惩罚函数值。当将它们用作拟合公式中的$\mathbf{B}$ 矩阵时，会得到不适定先验。只要观测数据足够多，从而得到正规后验分布，这就没有问题。

不同阶数的差分先验表现略有不同。图 2 展示了一个简单函数逼近任务的示例，其中使用了一阶、二阶和三阶差分。可以看出，尽管这些选择在数据范围内都能得到相似的拟合结果，但外推行为各不相同。一阶先验倾向于保持函数行为不变，二阶先验使其保持线性，而三阶先验则会进行二次外推。哪种方式最合适取决于具体应用。

请注意，有时在 GAM 中估计函数的最佳方案是混合使用不同阶数的差分先验；例如，我们可能知道，一维函数预期会沿一个方向线性外推，而在另一端则趋于某个常数值。

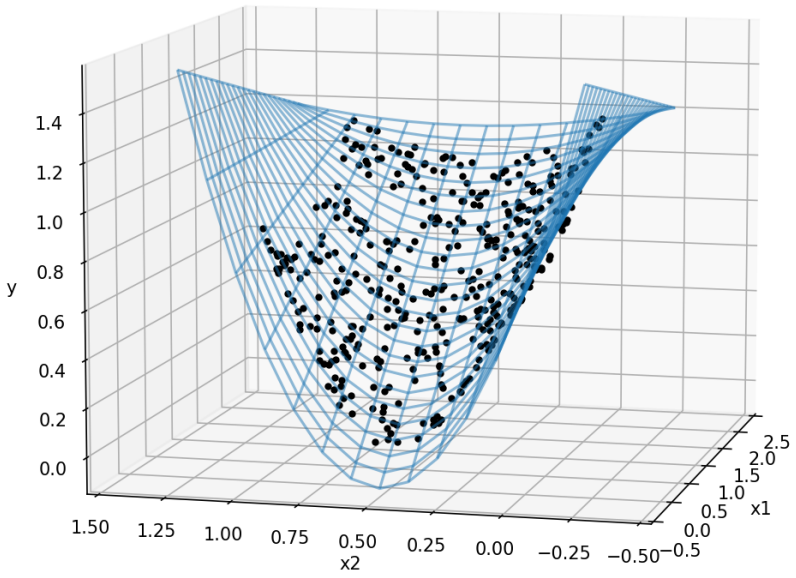

图 3：拟合数据　　围绕真实函数 $f(x_1,\ x_2) = 0.5x_1 + 4(x_2 - 0.5)^2/(1 + 2x_1)$ 使用 secon 生成　　d 阶差分先验　　第一维上的一阶差分先验，以及第二维上的三阶差分先验。

4.2 高维函数

样条基可以通过“张量积基”扩展到更高维，讨论见 [1]。这里我们聚焦于上一节讨论的一阶样条和差分先验。线性插值模型可以直接扩展到多个维度；需要确定合适的排序方式，按照所选排序将高维函数值向量化，并据此构造插值矩阵（计算正确向量索引之间的差值）。例如，采用列排序时，二维插值矩阵的第一行会有 4 个非零值，其形式如下：

$$\mathbf{A}_{1\cdot} = [0 \ \cdots \ 0 \ w_{11} \ w_{12} \ 0 \ \cdots \ 0 \ w_{13} \ w_{14} \ 0 \ \cdots \ 0], \tag{21}$$

其中，w_{1i}表示第一个观测值所在的“方形像素”各个角点的距离，且$\sum_i w_{1i} = 1$。N 维插值矩阵的每一行将有 2^N 个非零元素。

给定阶数的差分先验也可以用类似的方法扩展到更高维度。构造先验时，可以考虑分别为每个维度的方向定义先验（甚至可以为不同维度的差分指定不同的阶数）。例如，上述二维示例中两个方向的二阶差分矩阵如下：

$$\mathbf{D}_1 = \begin{bmatrix} -1 & 0 & 0 & 2 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 2 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 2 & 0 & 0 & -1 \end{bmatrix} \quad \mathbf{D}_2 = \begin{bmatrix} -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & -1 \end{bmatrix}. \tag{22}$$

此时，最终的先验矩阵 $\mathbf{B}$ 可以由两个维度上的先验组合而成。生成 N 维稀疏插值矩阵和差分先验矩阵的代码可在随附的软件包中获取[3]。

我们来看一个示例：拟合围绕真实函数 $f($ $x_1\ x_2$ $x_2 -$ $0.5\ ^2/\ 1 + 2x_1$ $x_1\ ,) = 0.5x_1 + 4(\ ) \ (\ )$ 随机生成的数据。

这里，该函数在第二个维度上是二次函数，其曲率随 的变化而变化；在第一个维度上则表现得近似线性。我们在此拟合一个函数，在第一个维度上采用二阶差分先验，在第二个维度上采用三阶先验。结果如图 3 所示；可以看到，拟合结果沿两个维度都能很好地外推。

在反问题领域，使用差分先验近似未知函数的方法已得到广泛研究。例如， [9] 讨论了差分先验在大气遥感中的应用，并展示了

[3]https://github.com/solbes/blinpy

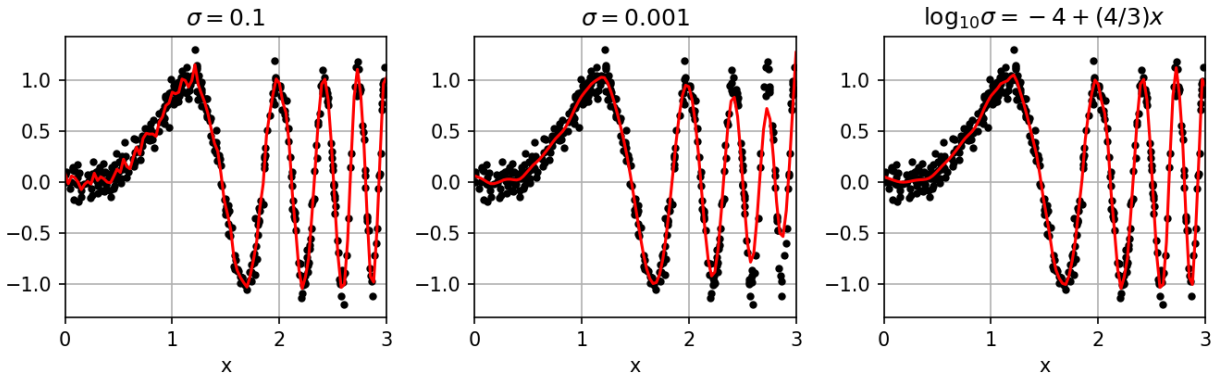

图 4：使用三种不同方式定义先验方差，对围绕真实函数 $f(x) = \sin(x^3)$ 生成的数据进行拟合。对于前两个图，先验方差固定为一个常数值；而对于最后一个图，先验方差则随 x 增大。

如何使先验与离散化无关。另请参见 [10, 11] 及其中引用的文献，了解关于高斯马尔可夫随机场及其在反问题背景下与差分先验之间联系的讨论。

4.3 空间变化的平滑度

局部基方法具有另一层灵活性，而这通常未在 GAM 方法中得到利用。通过调整差分先验的先验方差，可以轻松实现函数的空间变化平滑性。图 4 展示了一个例子，其目标是拟合一个平滑度明显非恒定的函数。对函数某一部分有效的先验方差，在其他部分可能效果很差。为函数的不同部分定义不同的方差即可解决这一问题。请注意，此处的方差行为是手动选择的。自动估计空间变化的平滑度参数颇为棘手，更多讨论请参见第 7 节。

空间变化平滑度的另一种典型应用，是对函数中的突变进行建模。通过在不连续点附近设置较大的先验方差，可以实现这种效果。参见 [12]，其中讨论了反问题的相关情形；例如，在医学成像应用中产生的边界保持先验，便是当前的热门研究课题。许多这类思路也可以推广到广义加性模型。图 5 给出了一个例子，其中数据包含一个较大的跳变；通过控制空间变化的平滑度，可以很好地拟合数据。

4.4 解决可识别性问题

采用局部基函数方法时，如果只对函数的导数进行惩罚，那么在同一模型中使用多个加性分量时，显然会出现可识别性问题。例如，对于两个加性分量，这些函数只能确定到一个常数：可以给第一个函数加上一个常数值，同时从第二个函数中减去该值，而模型预测和惩罚项（函数导数）都完全不变。因此，模型中需要更多先验信息才能使其可识别。另一种方法是在模型拟合中使用摩尔－彭若斯伪逆，从而得到该线性问题的最小范数解。

在贝叶斯框架中，解决这一问题的一个直接方法是为每个函数分别设置一个弱信息高斯先验（均值和方差），$\theta_i \sim N(\theta_i,\ \tau_i)$。函数的均值和方差可以手动设定；也可以先拟合一个假设函数为常数的模型，再允许其在这些值附近有所偏离（例如，可将标准差设为估计出的常数参数值的某个百分比）。

如果该模型不是变系数模型（即函数会与额外输入相乘），也可以加入一个共同的“平均系数” θ，并通过设置 $\theta_i \sim N(\theta, \tau)$ 要求所有估计函数都围绕该平均系数。平均系数可以与其他参数一起进行估计，也可以取数据的均值，例如。标准差可以取数据标准差的某个倍数，例如。请注意，选择 $\theta = 0$ 会得到该问题的最小范数解作为最大后验估计。

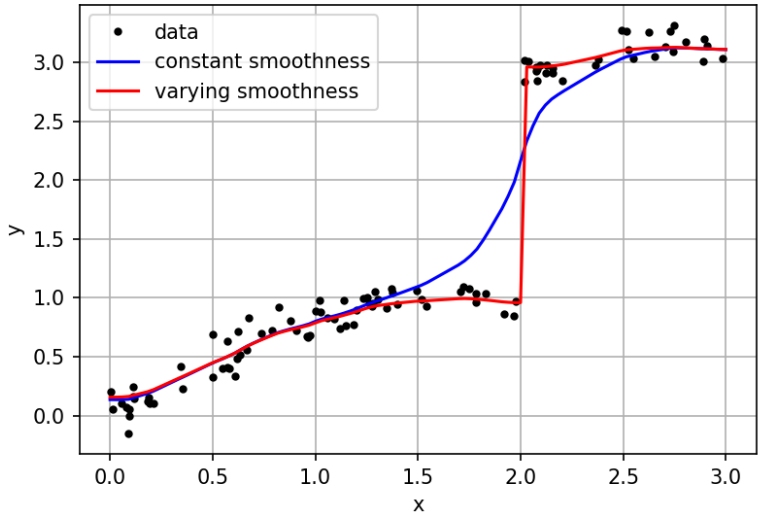

图5：通过 1）保持先验方差恒定，2）提高 $x = 2$ 附近的先验方差，对围绕真实函数 $f(x) = \sin(x) + \mathbb{1}(x > 2)$ 生成的随机数据进行拟合。

[1] 中用于解决该问题的方法，是向估计问题中加入约束，使得平滑分量的和应为零，然后向模型中加入一个截距项，以确定正确的水平。这样做的好处是，平滑函数的置信带会保持“较窄”。然而，这种方法不能直接用于变系数模型；需要加入若干个与相应输入相乘的常数参数。

请注意，第 3 节讨论的 GP 参数化不存在这一可识别性问题，因为 GP 先验直接惩罚函数值，并且每个函数都会趋向于各自选定的均值。

4.5 周期且对称的先验

在 GP 先验的情形下，只需选择具有所需周期性或对称性性质的合适核函数，即可直接实现周期性和对称性。采用局部基函数和差分先验方法时，则需要在差分先验矩阵中加入额外的行，才能获得此类行为。

只需在先验系统中增加一行，计算首个与最后一个函数值之差，即可实现函数值的周期性。不过，除了使边界处的函数值相匹配之外，人们通常还希望某些导数也相匹配。在一维情形下，若要求边界处的函数值和一阶导数相匹配，则需在先验系统中加入以下几行：

$$D_{periodic} = \begin{bmatrix} -1 & 0 & 0 & \cdots & 0 & 0 & 1 \\ 1 & -1 & 0 & \cdots & 0 & -1 & 1 \end{bmatrix}. \tag{23}$$

周期性假设的“强度”可以通过相关的先验方差进行控制。通过计算正确函数值之间的差异，可以类似地获得周期性先验的自定义周期。与高斯过程方法相比，这需要更多一些手动操作；在高斯过程方法中，周期是协方差核中的一个显式参数。不过话说回来，差分先验提供了更大的灵活性，因为无需明确指定任何具有限制性的协方差核。

除了周期性之外，对称性也是底层函数常见的假设。通过对对称轴两侧计算出的相应差分施加惩罚，可以强制实现对称性。例如，对于一个六维输入网格，可以通过将关于定义域中点的先验系统矩阵设为

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{24}$$

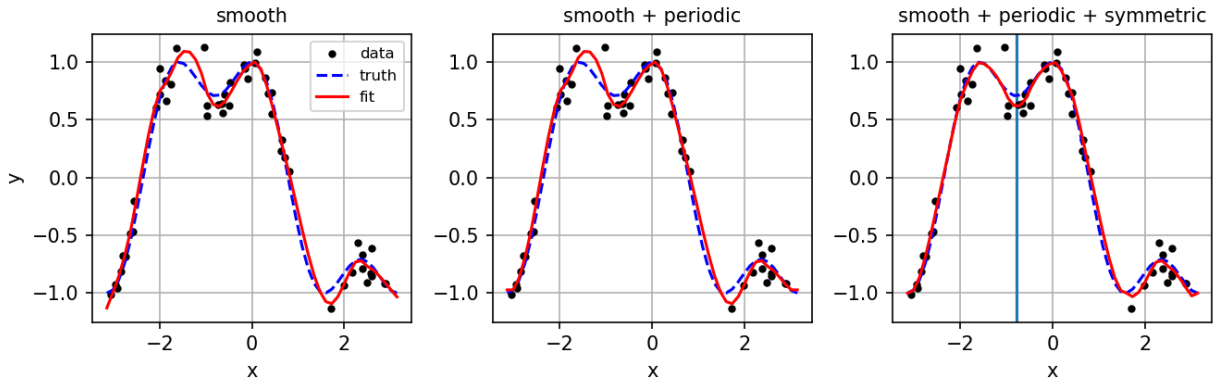

图6：用三种不同的先验拟合在范围 $x \in [-\pi, \pi]$ 内围绕真实函数 $f(x) = \cos^3(x) - \sin^3(x)$ 生成的数据。

图 6 给出了拟合对称周期函数的示例。在第一幅图中，我们仅使用光滑先验拟合模型。在第二幅图中，我们加入周期先验，可以看到定义域末端的值和导数相匹配。在最后一幅图中，我们加入了函数需要对称的先验，这进一步改善了拟合效果。请注意，最后一幅图中的对称性并不是关于定义域中心的。

5 单调函数

在某些应用中，我们可能希望要求函数（沿某个输入维度）单调递增或递减。这类先验信息无法直接纳入我们的线性高斯框架。纳入单调性约束的一种方法是将估计问题重新表述为受约束线性高斯问题。高效的二次规划方法（例如 [13],）可用于求解此类问题。

让我们考虑以下受约束线性高斯系统

$$\mathbf{y} = \mathbf{A}\boldsymbol{\theta} + \boldsymbol{\varepsilon} \tag{25}$$

$$\boldsymbol{B\theta} \sim \mathrm{N}\left(\boldsymbol{\mu}_{\mathrm{pr}}, \boldsymbol{\Gamma}_{\mathrm{pr}}\right) \tag{26}$$

$$\mathbf{C}\boldsymbol{\theta} \geq \mathbf{c}, \tag{27}$$

其中，最后一个方程给出了约束。例如，通过约束一阶差分为正，可以实现单调性。类似地，要求二阶导数为正即可实现凸性。问题中也可以包含等式约束。

上述系统并不完全符合可直接应用二次规划方法的形式，但将其转换为这种形式很简单。一般的二次规划问题形式为

$$\text{minimize} \quad \frac{1}{2}\boldsymbol{\theta}^T\mathbf{G}\boldsymbol{\theta} - \mathbf{a}^T\boldsymbol{\theta} \tag{28}$$

$$\text{so that} \quad \mathbf{K}^T\boldsymbol{\theta} \geq \mathbf{b}. \tag{29}$$

很容易验证，通过设$\mathbf{K} = \mathbf{C}^T$、$\mathbf{b} = \mathbf{c}$ 以及

$$\mathbf{G} = \mathbf{A}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{obs}}^{-1}\mathbf{A} + \mathbf{B}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{pr}}^{-1}\mathbf{B} \tag{30}$$

$$\mathbf{a} = \mathbf{A}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{obs}}^{-1}\mathbf{y} + \mathbf{B}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{pr}}^{-1}\boldsymbol{\mu}_{\mathrm{pr}}. \tag{31}$$

图 7 给出了一个拟合单调函数的简单示例。请注意，这种方法只能给出该问题的最大后验估计，而不能给出完整的后验分布或后验均值估计。

另一种纳入单调性信息的方法见 [14], ，该方法通过在选定的输入位置设置合成导数观测来实现单调性。不过，这种方法并不直接适用于广义加性模型语境。

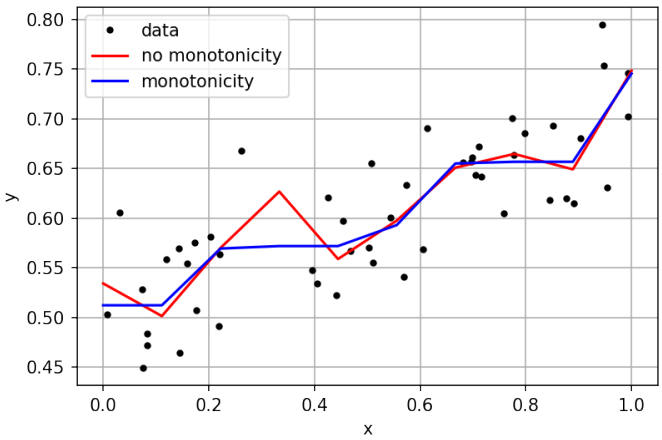

图7：拟合　　围绕 $y = (1 + x)/2$ 生成的随机数据，无论是否施加单调性约束。

6 全局基与局部基：优点与缺点

本文介绍了两种构建广义加性模型的方法：“全局”高斯过程方法和利用差分先验的“局部”方法。高斯过程方法的一个优点是，有多种核函数可供选择，相关文献也十分丰富。借助降维技巧，只要可以假设足够的光滑性，随着输入维度和数据量的增长，该方法在计算上也依然高效。此外，先验所产生的函数类型十分明确；从先验中采样很直接，可以轻松预先控制允许的函数值范围，而且不存在可辨识性问题，因为每个函数都有各自的均值。此外，高斯过程方法从构造上便具有离散化不变性。

采用降维的高斯过程方法还有一个优点：它可以嵌入非线性模型，而不会使估计问题的维度增加太多。例如，在低维问题中，对基函数权重进行马尔可夫链蒙特卡罗（MCMC）抽样会容易得多，如 [6, 15] 所示。

在高斯过程协方差函数中显式纳入周期性和对称性很容易，而局部方法则需要更多人工操作，才能恰当地设置差分矩阵。不过，差分先验在设置对称轴方面更灵活，例如便是如此。

高斯过程方法显而易见的缺点是其全局特性；它难以描述尖锐的局部特征，也难以实现空间变化的平滑度。此外，人们始终需要对函数值（均值）的水平作出判断，而在局部方法中，我们只需对导数施加惩罚，无须对函数值本身施加任何先验知识。另外，GP 方法的外推行为始终相同：在数据范围之外，函数会趋近于其先验。这可能并不理想，而采用局部方法则可以实现线性或二次外推等外推行为。

最后，我们想指出，选择合适的方法取决于具体应用，无法明确断言哪种方法优于另一种。

7 超参数估计

在许多问题中，建模者可能会满足于手动确定超参数（相关长度、差分先验方差、观测误差方差等）的取值。但一般来说，采用各种方法自动校准这些取值会很有帮助。本节将讨论广义加性模型中超参数估计的各种方法。

7.1 DirectMAP 最大后验估计

获取超参数值的一种直接方法，是在参数与超参数的联合空间中进行直接的最大后验优化。显然，这并未完整解决贝叶斯估计问题，因为它只给出了在超参数取最大后验估计时参数的后验分布，但在实践中，这种方法可以得到效果良好的估计。还要注意，我们可以利用该优化问题的结构，

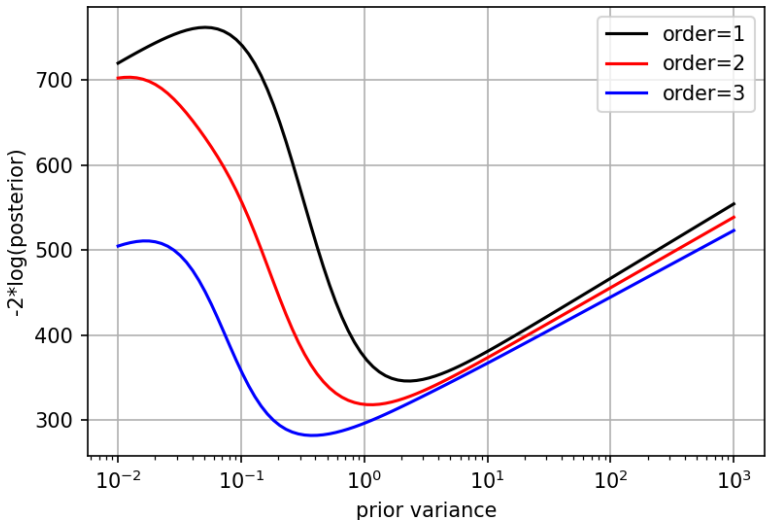

图 8：图 2 所示问题中，给定先验方差时的最小负对数后验密度值。

给定超参数后，对数后验是一个二次函数，其最优解可以通过解析方法求得。因此，我们只需在迭代优化循环中反复拟合模型。

例如，我们来考虑图 2 所示的简单拟合问题，其中我们使用了三种阶数的差分先验，并为每种阶数设置了四个不同的先验方差。在这里，我们针对每种阶数的一系列先验方差反复拟合模型，并计算拟合模型的对数后验值。结果如图 8 所示。可以看出，每种先验阶数对应的“最优”先验方差值都明显在 1 左右。

请注意，当先验方差趋近于零时，后验密度会出现病态最大值。先验方差为零会使先验模型（差分）与数据完美拟合，从而导致这种边界情况。可以通过为先验方差设置合理的限制（如图 8 的示例所示），或纳入有关方差参数的其他先验信息，来避免这一病态最优值。

7.2 最大化证据

最大化证据（又称边际似然）的思想，是计算先验分布下数据的似然，并将其最大化。为了推导先验预测分布，让我们回顾一下线性高斯系统：

$$\begin{aligned}\mathbf{y} &= \mathbf{A}\boldsymbol{\theta} + \varepsilon \\ \mathbf{B}\boldsymbol{\theta} &\sim \mathrm{N}\left(\boldsymbol{\mu}_{\mathrm{pr}}, \boldsymbol{\Gamma}_{\mathrm{pr}}\right).\end{aligned}$$

假设下方方程中的先验系统不是欠定的，我们可以将先验写为 $\boldsymbol{\theta} \sim N(\boldsymbol{\mu}_B,\ \boldsymbol{\Sigma}_B)$，其中

$$\boldsymbol{\mu}_B = \boldsymbol{\Sigma}_B \mathbf{B}^{\mathrm{T}} \boldsymbol{\Gamma}_{\mathrm{pr}}^{-1} \boldsymbol{\mu}_{pr} \tag{32}$$

$$\boldsymbol{\Sigma}_B^{-1} = \mathbf{B}^{\mathrm{T}} \boldsymbol{\Gamma}_{\mathrm{pr}}^{-1} \mathbf{B}. \tag{33}$$

因此，先验预测分布为

$$\mathbf{y} \sim \mathrm{N}\left(\mathbf{A}\left(\mathbf{B}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{pr}}^{-1}\mathbf{B}\right)^{-1}\mathbf{B}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{pr}}^{-1}\boldsymbol{\mu}_{pr},\ \mathbf{A}\left(\mathbf{B}^{\mathrm{T}}\boldsymbol{\Gamma}_{\mathrm{pr}}^{-1}\mathbf{B}\right)^{-1}\mathbf{A}^{\mathrm{T}} + \boldsymbol{\Gamma}_{\mathrm{obs}}\right). \tag{34}$$

注意，只有先验分布是适定的，才能计算上述结果；例如，当矩阵 $\boldsymbol{B}$ 为秩亏时，先验分布可能并不适定。如果我们只惩罚函数的某些导数，就会陷入这种情况。不过，我们始终可以为函数值添加一些非信息先验，从而使这种方法能够运行。

7.3 交叉验证与后验预测评分

交叉验证（CV）是贝叶斯模型（以及其他模型）中另一种常用的超参数调优方法。交叉验证通过留出一部分数据，并利用这部分数据评估拟合优度。拟合优度得分

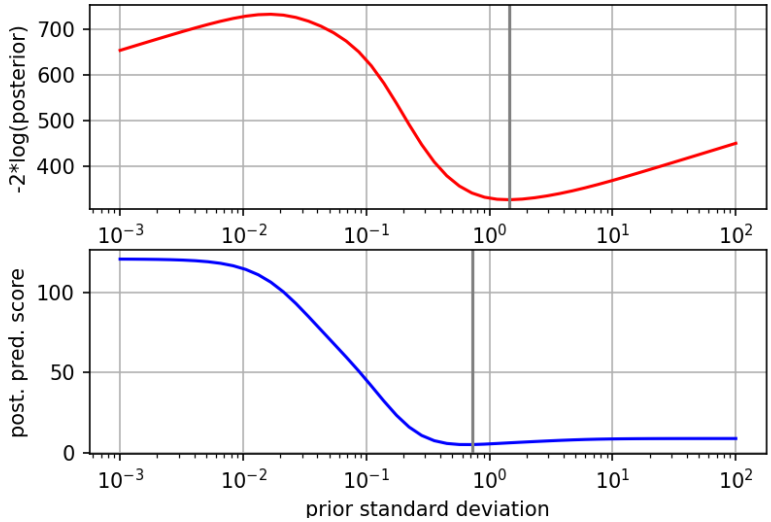

图 9：使用图 2 所示的简单拟合示例，采用二阶平滑先验，对比两种超参数校准方法。上图：直接最大后验估计；下图：留出 10 个数据点后计算得到的后验预测得分。

这里可以是模型预测值与数据之间偏差平方和等指标。在贝叶斯框架下，更可取的方法可能是使用“后验预测得分”，即在给定拟合模型的情况下，留出数据的似然。在我们的线性高斯设定中，新数据$\mathbf{y}*$ 的后验预测分布为

$$\mathbf{y}*|\mathbf{y} \sim \mathrm{N}\left(\mathbf{A}\boldsymbol{\mu}_{pos}, \mathbf{A}\boldsymbol{\Gamma}_{\mathrm{pos}}\mathbf{A}^{\mathrm{T}} + \boldsymbol{\Gamma}_{\mathrm{obs}}\right), \tag{35}$$

交叉验证得分就是在该分布下$\mathbf{y}*$ 的对数密度值。

请注意，根据公式 4，我们通常只能获得后验协方差的（通常是稀疏的）逆矩阵，因此上述计算可以通过求解线性方程组来完成。虽然$\mathbf{A}$ 和 $\boldsymbol{\Gamma}_{\mathrm{pos}}^{-1}$ 通常是稀疏的，但上述协方差是稠密的；当观测值数量 N 较大时，这会使计算变得困难，因为公式（35）中的协方差大小为 $N \times N$。

最后，与第 7.2 节讨论的证据最大化方法不同，交叉验证计算同样适用于先验不适定的情况。

图 9 给出了简单拟合问题中最大后验估计得分与后验预测得分的比较。两者给出的结果非常相似。请注意，由于先验系统矩阵$\mathbf{B}$是秩亏的，基于证据的方法在此处并不适用。

7.4 其他方法

如上所述，当超参数固定时，对数后验通常可以解析求得。如果为超参数采用了合适的（例如共轭）先验，那么超参数也可能具有同样的性质（例如在局部基函数的情形下）。这样便可以采用可能高效的分块优化和吉布斯采样方案来优化参数并从后验分布中获取样本。

另一种可能的方法是使用变分推断技术来估计参数和超参数。如果使用共轭先验，可以采用标准的变分贝叶斯更新来近似后验分布，相关讨论和该方法的 Python 实现请参见 [16] 。此外，INLA 方法 [3]提供了一种高效技术，可联合近似参数和超参数的后验分布。

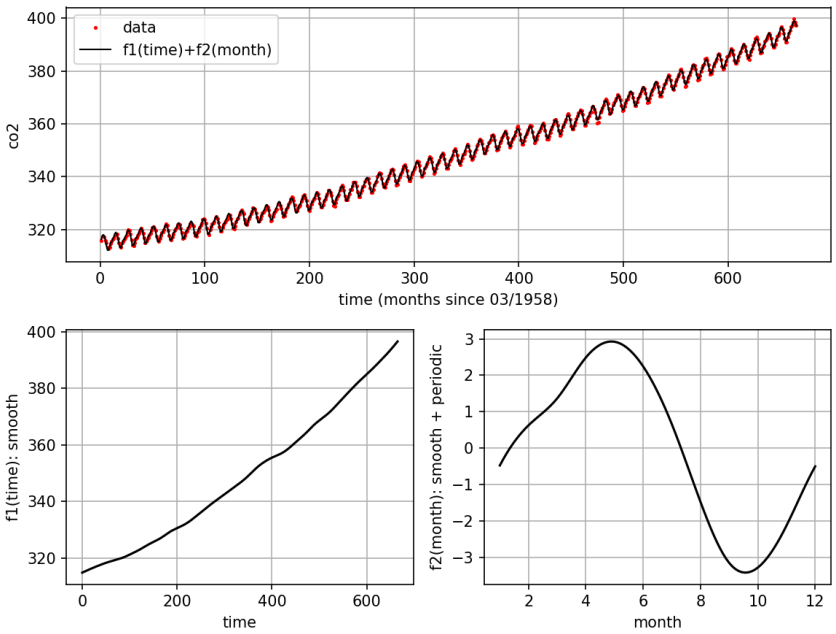

图10：上：数据（红色）与拟合模型（红色）。下：拟合函数 f_1(时间)和 f_2(月份)。

8 更多数值示例

前几节已经提供了若干关于此处所述方法的数值示例。本节将再给出几个数值示例。这些示例以及前面示例的代码均可在此处获取 [4]。

8.1 莫纳罗亚二氧化碳数据

这里，我们来看看著名的莫纳罗亚二氧化碳数据集，其中包含莫纳罗亚观测站 [5] 的二氧化碳浓度月平均测量值。数据呈现出明显的平滑时间趋势，以及年度周期分量。这启发我们建立一个模型

$$y = f_1(\text{time}) + f_2(\text{month}) + \varepsilon,$$

其中， f_1 和 f_2 是平滑函数，后者具有周期性。

首先，我们使用局部基函数方法，并手动选择平滑度参数来拟合数据。所选参数的详情请参阅附带的代码。结果见图10。可以看到，模型很好地拟合了数据，并估计出了外观良好的平滑趋势。

接下来，我们最大化两条平滑函数的平滑度参数的似然。结果见图11。与之前的拟合相比，函数 f_1 变得稍微平滑了一些，但除此之外，拟合结果相似。

8.2 合成示例：超参数调优

为了展示各种超参数学习方法，我们在此考虑一个由[1]修改而来的合成示例。我们通过向真实函数添加高斯噪声，在区间 $[x, y]$ 内生成随机数据

$$f_{true}(x) = 2\sin(\pi x) + \exp(2x) + x^{11}(10(1-x))^6/5 + 10^4 x^3(1-x)^{10},$$

实验详情请参阅附带的代码。

也就是说，我们有三个平滑度等级不同的加性函数，手动调整这些平滑度等级需要付出大量努力。在这里，我们拟合一个模型

$$y = f_1(x) + f_2(x) + f_3(x) + \varepsilon$$

[4] https://github.com/solbes/gam_论文_示例 [5] Pieter Tans 博士，美国国家海洋和大气管理局全球监测实验室（https://gml.noaa.gov/ccgg/trends/）以及 Ralph Keeling 博士，斯克里普斯海洋研究所（https://scrippsco2.ucsd.edu）。

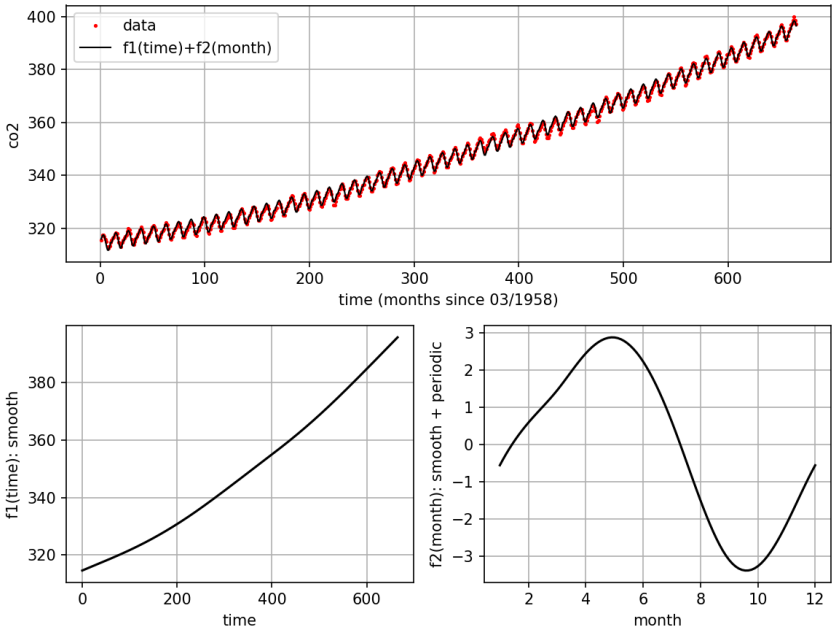

图11：上：数据（红色）与拟合模型（红色）。下排：拟合函数 f_1（时间）和 f_2（月份）。

采用局部基函数方法，并使用第7节介绍的超参数调优方法估计这三个函数的光滑度等级。各个分量的数据和函数拟合结果如图12所示。请注意，函数的形状得到了很好的恢复，但函数值本身的水平无法从数据中识别出来。还请注意，由于使用了非正规先验，本示例无法采用基于证据的方法。

9 结论

广义加性模型（GAMs）是拟合“半参数”模型的灵活工具，它将参数技术与非参数技术结合起来。本文讨论了广义加性模型的两种不同形式：“全局基函数方法”，其中广义加性模型中的非参数函数通过高斯过程（GPs）来描述；以及“局部基函数方法”，其中函数值直接作为模型中的未知参数处理，再通过高斯先验进行惩罚，从而使函数达到适当的平滑度等级。本文还讨论了模型拟合中的各种数值方面，包括降维、空间变化的平滑度、单调函数和超参数调优方法。

哪种方法最适合实际建模任务，取决于具体问题。局部基函数方法在设计函数形状方面更加灵活，但需要投入更多人工工作，以正确设置先验矩阵。若能应用降维技巧，全局高斯过程方法的计算效率会非常高，而且更容易嵌入非线性模型。关于这两种方法的优缺点，详见第6节。

最后补充一点，本研究还开发了两个用于拟合此类广义加性模型的 Python 软件包。第一个软件包 gammy[6], 采用全局（高斯过程）基函数方法来拟合广义加性模型。另一个软件包 blinpy[7],是一个通用工具，可用于拟合第1.1节所述形式的贝叶斯线性高斯系统，并包含用于构建系统和先验矩阵的各种工具。示例请参阅软件包页面。

[6]https://github.com/malmgrek/gammy[7]
https://github.com/solbes/blinpy

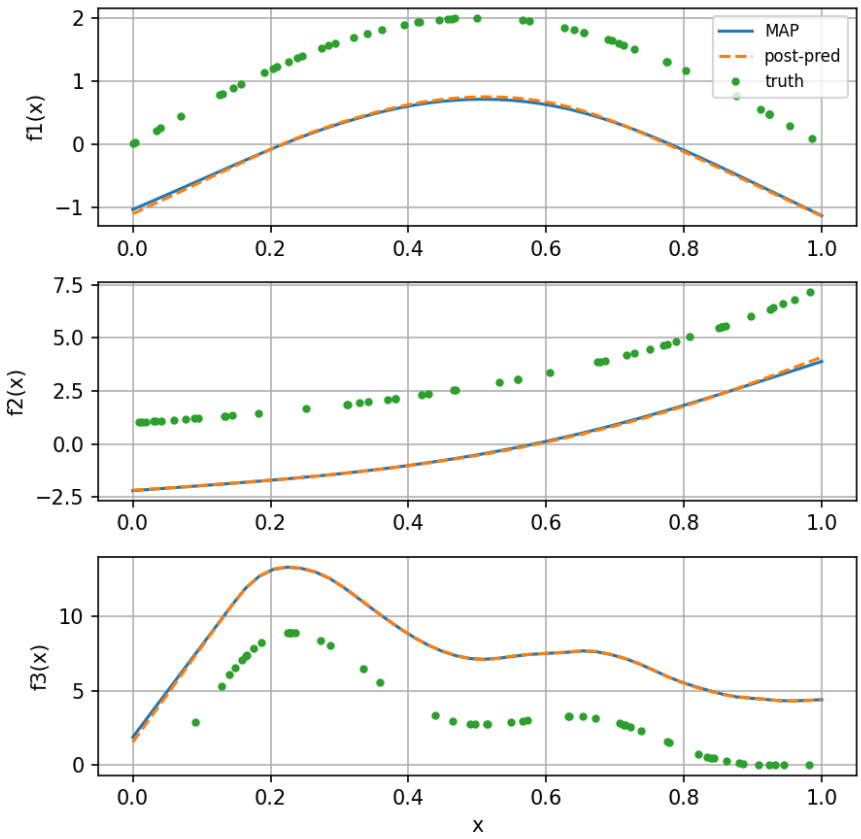

图12：使用局部基函数方法同时估计模型的三个分量函数。采用两种不同的调参方法估计平滑度级别超参数：组合最大后验和后验预测最大化。

参考文献

[1] Simon Wood。《广义加性模型：R语言入门》，第二版。Chapman & Hall/CRC统计科学丛书。博卡拉顿：CRC出版社/泰勒与弗朗西斯集团，2017年。[2] Simon N. Wood。广义加性模型及其扩展的推断与计算。*TEST，29*：*307–339*，2020年。[3] Håvard Rue、Sara Martino 和 Nicolas Chopin。利用积分嵌套拉普拉斯近似对潜在高斯模型进行近似贝叶斯推断。英国皇家统计学会会刊：*B*辑（统计方法学），71(2)：319–392，2009年。[4] Tibshirani R.、Hastie T. 和 Friedman J.H.《统计学习要素》，第2版。施普林格，2009年。[5] Rasmussen C.E. 和 Williams C.《机器学习中的高斯过程》。麻省理工学院出版社，2006年。[6] Y. Marzouk 和 H. Najm。反问题中贝叶斯推断的降维与多项式混沌加速。计算物理学杂志，*228，1862–1902*，2009年。[7] W. Vetterling、W. Press、S. A. Teukolsky 和 B. Flannery。《C语言数值算法：科学计算的艺术》，第2版。剑桥大学出版社，英国剑桥，1992年。[8] Laub A.J.《科学家与工程师的矩阵分析》。工业与应用数学学会，2005年。[9] Haario H.、Laine M.、Lehtinen M.、Saksman E. 和 Tamminen J. 用于遥感高维反演的马尔可夫链蒙特卡罗方法。英国皇家统计学会会刊*B*辑，*66*，第*3*部分，第*591–607*页，2004年。[10] Johnathan M. Bardsley。用于反问题的高斯马尔可夫随机场先验。逆问题与成像，7(2)：397–416，2013年。

[11] H. Rue 和 L. Held。《高斯马尔可夫随机场：理论与应用》（第*1*版）。Chapman and Hall/CRC.，2005年。[12] Johnathan M Bardsley、Daniela Calvetti 和 Erkki Somersalo。用于 PET图像 保边重建的分层正则化。《反问题》，26(3):035010，2010年2月。[13] D. Goldfarb 和 A. Idnani。求解严格凸二次规划的一种数值稳定的对偶方法。《数学规划》，*27, 1–33*，1983年。[14] Jaakko Riihimäki 和 Aki Vehtari。包含单调性信息的高斯过程。载于第十三届国际人工智能与统计学会议论文集，*PMLR*，第9:645–652 页，2010年。[15] S Tukiainen、J Railo、M Laine、J Hakkarainen、R Kivi、P Heikkinen、H Chen 和 J Tamminen。利用降维和马尔可夫链蒙特卡罗，从傅里叶变换红外数据中反演大气甲烷剖面。《地球物理研究杂志：大气》，121(7):10312–10327，2016年。[16] Jaakko Luttinen。Bayespy：Python 中的变分贝叶斯推断。《机器学习研究杂志》*17*：*41:1-41:6*，2016年。